5   **VIDEO COMPRESSION USING ADAPTIVE SELECTION OF GROUPS OF FRAMES, ADAPTIVE BIT ALLOCATION, AND ADAPTIVE REPLENISHMENT**

10

PRIORITY AND RELATED APPLICATIONS

15   The present application claims priority to provisional patent application entitled, "Video Processing Method with General and Specific Applications," filed on July 11, 2000 and assigned U.S. Application Serial No. 60/217,301. The present application is also related to non-provisional application entitled, "Adaptive Edge Detection and Enhancement for Image Processing," (attorney docket number 07816-105003) filed on July 11, 2001 and assigned U.S.

20 Application Serial No. _____; and non-provisional application entitled, and non-provisional application entitled, "System and Method for Calculating an Optimum Display Size for a Visual Object," (attorney docket number 07816-105002) filed on July 11, 2001 and assigned U.S. Application Serial No. _____.

25 FIELD OF THE INVENTION

    The present invention relates to the processing of a video stream and more specifically relates to the improvement of video stream compression by adaptively selecting a group of pictures based on video stream content, by adaptively allocating bits to generate a compressed video stream, and by adaptively replenishing

30 macroblocks.

BACKGROUND OF THE INVENTION

    Recent advancements in communication technologies have enabled the

35 widespread distribution of data over communication mediums such as the Internet and broadband cable systems. This increased capability has lead to increased demand for the distribution of a diverse range of content over these communication mediums. Whereas early uses of the Internet were often limited to the distribution of raw data,

5   more recent advances include the distribution of HTML-based graphics and audio files.

More recent efforts have been made to distribute video media over these communication mediums. However, because of the large amount of data needed to represent a video presentation, the data is typically compressed prior to
10   distribution. Data compression is a well-known means for conserving transmission resources when transmitting large amounts of data or conserving storage resources when storing large amounts of data. In short, data compression involves minimizing or reducing the size of a data signal (e.g., a data file) in order to yield a more compact digital representation of that data signal. Because digital representations of audio and
15   video data signals tend to be very large, data compression is virtually a necessary step in the process of widespread distribution of digital representations of audio and video signals.

Fortunately, video signals are typically well suited for standard data compression techniques. Most video signals include significant data redundancy.
20   Within a single video frame (image), there typically exists significant correlation among adjacent portions of the frame, referred to as spatial correlation. Similarly, adjacent video frames tend to include significant correlation between corresponding image portions, referred to as temporal correlation. Moreover, there is typically a considerable amount of data in an uncompressed video signal that is irrelevant. That
25   is, the presence or absence of that data will not perceivably affect the quality of the output video signal. Because video signals often include large amounts of such redundant and irrelevant data, video signals are typically compressed prior to transmission and then decompressed again after transmission.

Generally, the distribution of a video signal includes a transmission
30   unit and a receiving unit. The transmission unit will receive a video signal as input and will compress the video signal and transmit the signal to the receiving unit. Compression of a video signal is usually performed by an encoder. The encoder typically reduces the data rate of the input video signal to a level that is predetermined by the capacity of the transmission medium. For example, for a typical video file
35   transfer, the required data rate can be reduced from about 30 Megabits per second to

2

5   about 384 kilobits per second. The compression ratio is defined as the ratio between the size of the input video signal and the size of the compressed video signal. If the transmission medium is capable of a high transmission rate, then a lower compression ration can be used. On the other hand, if the transmission medium is capable of a relatively low transmission rate, then a lower compression ratio can be used.

10   After the receiving unit receives the compressed video signal, the signal must be decompressed before it can be adequately displayed. The decompression process is performed by a decoder. In some applications, the decoder is used to decompress the compressed video signal so that it is identical to the original input video signal. This is referred to as lossless compression, because no data is lost

15   in the compression and decompression processes. The majority of encoding and decoding applications, however, use lossy compression, wherein some predefined amount of the original data is irretrievably lost in the compression and expansion process. In order to decompress the video stream to its original (pre-encoding) data size, the lost data must be replaced by new data. Unfortunately, lossy compression of

20   video signals will almost always result in the degradation of the output video signal when displayed after decoding, because the new data is usually not identical to the lost original data. Video signal degradation typically manifests itself as a perceivable flaw in a displayed video image. These flaws are typically referred to as noise. Well-known kinds of video noise include blockiness, mosquito noise, salt-and-pepper

25   noise, and fuzzy edges. The data rate (or bit rate) often determines the quality of the decoded video stream. A video stream that was encoded with a high bit rate is generally a higher quality video stream than one encoded at a lower bit rate.

        Conventional methods of compressing video signals include the partitioning of the video signal into groups of pictures. Unfortunately, conventional

30   compression techniques utilize inefficient and arbitrarily simple methods of grouping pictures that result in higher output signal bit rates and/or lower output signal quality. Moreover, because these conventional techniques use arbitrarily simple picture groupings, they do not provide the opportunity to maximize the output signal quality by appropriately allocating bits among pictures and picture groups in the output

35   signal. Finally, these compression techniques typically apply compression methods

3

5   that result in the propagation and amplification of noise, especially in background potions of a video picture.

Therefore, there is a need in the art for video signal compression that efficiently groups pictures in a video stream and provides for lower output signal bit rates and higher output signal quality. The video signal compression also should

10   maximize the output signal quality by appropriately allocating bits among pictures and picture groups in the output signal. In addition, the video signal compression also should apply compression methods that reduce noise in the output signal. Finally, the method should enable the use of various sampling techniques and should enable the selection of an output stream, based on the sampling technique providing the best

15   video stream.

SUMMARY OF THE INVENTION

The present invention provides video signal compression that efficiently groups pictures in a video stream into variably-sized groups of pictures

20   (GOPs) thereby providing lower achievable output signal bit rates and higher output signal quality. The video signal compression maximizes the output signal quality by appropriately allocating bits among pictures and picture groups in the output signal. An adaptive method of bit allocation among picture groups and within the pictures in those picture groups enables the efficient allocation of bits, according to the relative

25   sizes of the picture groups. The video signal compression of the present invention also applies compression methods that reduce noise in the output signal, by utilizing a macroblock-based tunable conditional replenishment technique. The conditional replenishment technique exploits the similarities among images in the variably-sized GOPs to further minimize output bit rate and maximize the output signal quality. An

30   analysis-by-synthesis method is also provided to select a best asynchronous sampling method among candidate sampling procedures.

In one aspect of the invention, a method is provided for processing an input video stream comprising a series of pictures. A first scene change is detected between a first scene in the input video stream and a second scene in the input video

4

5   stream. The method classifies the first picture following the first scene change as an
intra-picture (I-picture).

In another aspect of the invention, the input stream processing method
determines whether there are a predetermined number of pictures between the first I-
picture and a second scene change. A second picture in the input video stream is

10   classified as a second I-picture, where it is determined that the predetermined number
of pictures exist between the first intra-picture and the second scene change, wherein
the second picture coincides with the predetermined number of pictures.

In yet another aspect of the invention, a system is provided for
organizing a series of pictures in an input video stream into at least one group of

15   pictures (GOP). The system includes a picture grouping module for detecting a scene
change in the series of pictures and for classifying a first picture following the scene
change as a first intra-picture (I-picture). The picture grouping module also can
classify at least one other picture following the scene change as a predicted picture (P-
picture) and can classify at least one second picture as a bi-directionally predicted

20   picture (B-picture). The system also includes a bit allocation module for determining
whether a first GOP uses less than a predetermined target number of bits and further
operative to allocate an unneeded bit to a second GOP in response to a determination
that the first GOP uses less than the predetermined target number of bits.

The various aspects of the present invention may be more clearly understood

25   and appreciated from a review of the following detailed description of the disclosed
embodiments and by reference to the drawings and claims.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram depicting an exemplary video stream

30   comprised of a series of video pictures.

Figure 2 is a flowchart depicting an exemplary method for coding,
transmitting, and decoding a video stream.

Figure 3 is a block diagram depicting a system for encoding a video
stream that is an exemplary embodiment of the present invention.

5   Figure 4 depicts a conventional decoding system for receiving an encoded video stream and providing decoded video and audio output.

Figure 5 is a block diagram depicting an exemplary selection of picture encoding modes in a GOP.

Figure 6 is a block diagram depicting an exemplary timeline

10   comparing the occurrence of scene changes in a video stream with alternative GOP size formats.

Figure 7 is a flowchart depicting an exemplary method for creating GOPs of varying sizes.

Figure 8 is a graph depicting a typical relationship between the bits

15   generated by a conventional compression method and a conventional group of pictures.

Figure 9 is a series of block diagrams and graphs comparing the generated bit graph of a conventional compression method with a generated bit graph of an exemplary embodiment of the present invention.

20   Figure 10a is a flow chart depicting an exemplary method for adaptively allocating bits among variable-sized groups of pictures.

Figure 10b is a flow chart depicting an exemplary method for adaptively allocating bits among pictures within a GOP.

Figure 11 is a simplified illustration depicting successive pictures in an

25   exemplary GOP divided into macroblocks.

Figure 12 is a flowchart depicting an exemplary method for performing conditional replenishment on a macroblock-basis.

Figure 13 is a flowchart depicting an exemplary method for generating and selecting between two sampling methods.

30

DETAILED DESCRIPTION OF THE EXEMPLARY EMBODIMENTS

The present invention provides video signal compression that efficiently groups pictures in a video stream into variably-sized groups of pictures (GOPs) thereby providing lower achievable output signal bit rates and higher output

35   signal quality. The video signal compression maximizes the output signal quality by

5   appropriately allocating bits among pictures and picture groups in the output signal.
An adaptive method of bit allocation among picture groups and within the pictures in
those picture groups enables the efficient allocation of bits, according to the relative
sizes of the picture groups. The video signal compression of the present invention
also applies compression methods that reduce noise in the output signal, by utilizing a

10   macroblock-based tunable conditional replenishment technique. The conditional
replenishment technique exploits the similarities among images in the variably-sized
GOPs to further minimize output bit rate and maximize the output signal quality. An
analysis-by-synthesis method is also provided to select a best asynchronous sampling
method among multiple non-uniform and/or uniform sampling procedures.

15

An Exemplary Operating Environment

Figure 1 is a block diagram depicting an exemplary video stream
comprised of a series of video pictures. A video stream is simply a collection of
related images that have been connected in a series to create the perception that

20   objects in the image series are moving. Because of the large number of separate
images that are required to produce a video stream, it is common that the series of
images will be digitized and compressed, so that the entire video stream requires less
space for transmission or storage. The process of compressing such a digitized video
stream is often referred to as "encoding." Among other things, encoding a video

25   stream typically involves removing the irrelevant and/or redundant digital data from
the digitized video stream. Once the video stream has been so compressed, a video
stream must usually be decompressed before it can be properly rendered or displayed.

The video stream **100** depicted in Figure 1 includes six, separate
images or pictures **102-112**. Typically, a video stream is displayed to a viewer at

30   about 30 frames per second. Therefore, the video stream **100** depicted in Figure 1
would provide about 0.2 seconds of playback at the typical display rate.

Generally, there is little noticeable change from one picture in the
series to the next. If a video stream were to be stored or transmitted without
compression, large amounts of redundant data would be stored because of the

35   significant video data overlap from one frame to the next. For video stream storage,

7

5   the storage of such redundant data is consumptive of memory resources. For video
stream transmission, the transmission of such redundant data significantly increases
transmission time and may be impossible at certain data transmission rates.

Video stream compression is one means for reducing the size of a
video stream. In short, video stream compression involves the elimination of
10   irrelevant and/or redundant video data from the video stream. Moreover, many
compression methods store only enough video data on a frame-by-frame basis to
represent the differences between one frame to the next. For example, many
compression methods store an intra-picture (I-Picture) that includes all or most of the
video data for a particular frame/picture in a video stream. Subsequent pictures can
15   be represented by predicted pictures (P-pictures) or by bi-directionally predicted
pictures (B-pictures). P-pictures are encoded using motion-compensated prediction
from a previous I-Picture or a previous P-Picture. B-pictures are encoded using
motion-compensation prediction from either previous or subsequent I-pictures or P-
pictures. B-pictures are not used in the prediction of other B-pictures or other P-
20   pictures. Accordingly, I-pictures require the most amount of video data and can be
compressed the least. P-pictures require less video data than I-pictures and can be
significantly compressed. B-pictures require the least amount of video data and can
be compressed the most.

In the example of Figure 1, the first picture **102** is an I-Picture.
25   Accordingly, much of the video data of the image of the first picture **102** would be
used to represent the first picture **102**. The second picture **104** may be a B-Picture
and, thus, may be represented in terms of video data differences with the I-Picture
102. Because the B-Picture **104** is bi-directionally predicted, it may also be presented
in terms of differences with the P-Picture **106**. The P-Picture **106**, in turn, is predicted
30   in terms of differences with the I-Picture **102**. The P-Picture **106** is not represented in
terms of differences with the B-Picture **104**.

Differences between video pictures are often predicted based on
calculated motion vectors. Motion vectors are well-known mathematical
representations of the movement and/or expected movement of visual "objects" in a
35   series of pictures in a video stream. In order to track and predict the motion of

5   objects, pictures are divided into picture elements (pels). Pels may be a video pixel or
some other definable division of a picture. In any event, object motion can be tracked
by reference to corresponding pels in a series of related video pictures.

Often, a video picture (or other digitized picture) is encoded as a
collection of blocks **116**. Each block is typically an 8-by-8–square of pels. In
10   addition, video pictures also are commonly divided into macroblocks that usually
contain 6 blocks (4 blocks for luminance and 2 blocks for chrominance signal).
Those skilled in the art will appreciate that the division of video pictures into blocks
and macroblocks is arbitrary, but helpful to the creation of video compression
standards. Moreover, the division of pictures into such blocks enables the
15   representation of P-pictures and B-pictures in terms of other pictures in the video
stream. This block/macroblock-based representation facilitates picture comparisons,
based on corresponding portions of successive pictures. As described above, this
representation further facilitates the compression of a video stream.

Fig. 2 is a flowchart depicting an exemplary method for coding,
20   transmitting, and decoding a video stream. One application for which the described
exemplary embodiment of the present invention is particularly suited is that of video
stream processing. Because of the large number of separate images that are required
to produce a video stream, it is common that the series of images will be digitized and
compressed (encoded), so that the entire video stream requires less space for
25   transmission or storage. Once the video stream has been so compressed, the video
stream must usually be decompressed before it can be properly displayed. The flow
chart of Fig. 2 depicts the steps that are generally followed to encode, decode, and
display a video stream.

The method of Fig. 2 begins at start block **200** and proceeds to step
30   **202**. At step **202**, the input video stream is prepared for encoding. Step **202** may be
performed by an encoder or prior to sending the video stream to an encoder. In any
event, the video stream can be modified to facilitate encoding. Indeed various
exemplary embodiments of the present invention are directed to various aspects of
performing this step. The following Figures and accompanying text are drawn to
35   describing those embodiments.

The method proceeds from step **202** to step **204**. At step **204**, the input video stream is encoded. As described, the encoding process involves, among other things, the compression of the digitized data making up the input video stream. For the purposes of this description, the terms "encoding" and "compression" are used interchangeably. Once the video stream has been encoded, it can be transmitted or stored in its compressed form. At step **206**, the encoded video bit stream is transmitted. Often this transmission can be made over conventional broadcast infrastructure, but could also be over broadband communication resources and/or internet-based communication resources.

The method proceeds from step **206** to step **208**. At step **208**, the received, encoded video stream is stored. As described above, the compressed video stream is significantly smaller than the input video stream. Accordingly, the storage of the received, encoded video stream requires fewer memory resources than storage of the input video stream would require. This storage step may be performed, for example, by a computer receiving the encoded video stream over the Internet. Those skilled in the art will appreciate that step **208** could be performed a variety of well-known means and could be even be eliminated from the method depicted in Fig. 2. For example, in a real-time streaming video application, the video stream is typically not stored prior to display.

The method proceeds from step **208** to step **210**. At step **210**, the video stream is decoded. Decoding a video stream includes, among other things, expanding (decompressing) the encoded video stream to its original data size. That is, the encoded video stream is expanded so that it is the same size as the input video stream. The irrelevant and/or redundant video data that was removed in the encoding process is replaced with new data. Various, well-known algorithms are available for decoding an encoded video stream. Unfortunately, these algorithms are typically unable to return the encoded video stream to its original form without some image degradation. Consequently, a decoded video stream is typically filtered by a post-processing filter to reduce flaws (e.g., noise) in the decoded video stream.

Once the video stream has been decoded, it is suitable for displaying. The method of Fig. 2 proceeds from step **210** to step **212** and the enhanced video stream is displayed. The method then proceeds to end block **214** and terminates.

An Exemplary Encoding System

Figure 3 is a block diagram depicting a system for encoding a video stream that is an exemplary embodiment of the present invention. The encoding system **300** receives a video input signal **302** and an audio input signal **304**. The video input **302** is typically a series of digitized images that are linked together in series. The audio input **304** is simply the audio signal that is associated with the series of images making up the video input **302**.

The video input **302** is first passed through a pre-processing filter **306** that, among other things, filters noise from the video input **302** to prepare the input video stream for encoding. The input video stream is then passed to the video encoder **310**. The video encoder compresses the video signal by eliminating irrelevant and/or redundant data from the input video signal. The video encoder **310** may reduce the input video signal to a predetermined size to match the transmission requirements of the encoding system **300**. Alternatively, the video encoder **310** may simply be configured to minimize the size of the encoded video signal. This configuration might be used, for example, to maximize the storage capacity of a storage medium (e.g., hard drive).

In a similar fashion, the audio input **304** is compressed by the audio encoder **308**. The encoded audio signal is then passed with the encoded video signal to the video stream multiplexer **312**. The video stream multiplexer **312** combines the encoded audio signal and the encoded video signal so that the signals can be separated and played-back substantially simultaneously. After the encoded video and encoded audio signals have been combined, the encoding system outputs the combined signal as an encoded video stream **314**. The encoded video stream **314** is thus prepared for transmission, storage, or other processing as needed by a particular application. Often, the encoded video stream **314** will be transmitted to a decoding system that will decode the encoding video stream **314** and prepare it for subsequent display.

11

5   In an exemplary embodiment of the present invention, the video input stream **302** can be further processed prior to encoding. In addition to the pre-processing performed by the pre-processing filter **306**, the exemplary encoding system **300** can prepare the input video stream **302** for encoding by generating a control signal for the input video stream to facilitate compression. For example, a rate

10   controller **320** can be used to match the output bit rate of the encoder to the capacity of transmission channel or storage device. Furthermore, The rate controller **320** can be used to control the output video quality. For efficient rate control, the exemplary encoding system **300** includes a picture grouping module **316**, a bit allocation module **318** and a bit rate controller **320**.

15   The picture grouping module **316** can process a video input stream by selecting and classifying I-pictures in the video stream. The picture grouping module **316** can also select and classify P-pictures in the video stream. As is discussed in more detail below, the picture grouping module **316** can significantly improve the quality of the encoded video stream. Conventional encoding systems arbitrarily select

20   I-pictures, by adhering to fixed-size picture groups. The exemplary coding system **300** can adaptively select I-pictures to maximize the encoded video stream quality.

The bit allocation module **318** can be used to enhance the quality of the encoded video bit stream by adaptively allocating bits among the groups of pictures defined by the picture grouping module **316** and by allocating bits among the pictures

25   within a given group of pictures. Whereas conventional decoding systems often allocate bits in an arbitrary manner, the allocation module **318** can reallocate bits from the picture groups requiring less video data to picture groups requiring more video data. Consequently, the quality of the encoded video bit stream is enhanced by improving the quality of the groups of pictures requiring more video data for high

30   quality representation.

The bit rate controller **320** uses an improved method of conditional replenishment to further reduce the presence of noise in an encoded video bit stream. Conditional replenishment is a well-known aspect of video data compression. In conventional encoding systems, a picture element or a picture block will be encoded

35   in a particular picture if the picture element or block has changed when compared to a

5   previous picture. Where the picture element or block has not changed, the encoder will typically set a flag or send an instruction to the decoder to simply replenish the picture element or block with the corresponding picture element or block from the previous picture. The bit rate controller **320** of an exemplary embodiment of the present invention instead focuses on macroblocks and may condition the

10   replenishment of a macroblock on the change of one or more picture elements and/or blocks within the macroblock. Alternatively, the bit rate controller **320** may condition the replenishment of a macroblock on a quantification of the change within the macroblock (e.g., the average change of each block) meeting a certain threshold requirement. In any event, the objective of the bit rate controller **320** is to further

15   reduce the presence of noise in video data and to simplify the encoding of a video stream.

A Conventional Decoding System

Figure 4 depicts a conventional decoding system for receiving an

20   encoded video stream and providing decoded video and audio output. The decoding system **400** receives an encoded video stream **402** as input to a video stream demultiplexer **404**. The video stream demultiplexer separates the encoded video signal and the encoded audio signal from the encoded video stream **402**. The encoded video signal is passed from the video stream demultiplexer **404** to the video decoder

25   **406**. Similarly, the encoded audio signal is passed from the video stream demultiplexer **404** to the audio decoder **410**. The video decoder **406** and a audio decoder **410** expand the video signal and the audio signal to a size that is substantially identical to the size of the video input and audio input described above in connection with Figure 3. Those skilled in the art will appreciate that various well-known

30   algorithms and processes exist for decoding an encoded video and/or audio signal. It will also be appreciated that most encoding and decoding processes are lossy, in that some of the data in the original input signal is lost. Accordingly, the video decoder **406** will reconstruct the video signal with some signal degradation, which is often perceivable as flaws in the output image.

The post-processing filter **408** is used to counteract noise found in a decoded video signal that has been encoded and/or decoded using a lossy process. Examples of well-known noise types include mosquito noise, salt-and-pepper noise, and blockiness. The conventional post-processing filter **408** includes well-known algorithms to detect and counteract these and other known noise problems. The post-processing filter **408** generates a filtered, decoded video output **412**. Similarly, the audio decoder **410** generates a decoded audio output **414**. The video output **412** and the audio output **414** may be fed to appropriate ports on a display device, such as a television, or may be provided to some other display means such as a software-based media playback component on a computer. Alternatively, the video output **412** and the audio output **414** may be stored for subsequent display.

As described above, the video decoder **406** decompresses or expands the encoded video signal **402**. While there are various well-known methods for encoding and decoding a video signal, in all of the methods, the decoder must be able to interpret the encoded signal. The typical decoder is able to interpret the encoded signal received from an encoder, as long as the encoded signal conforms to an accepted video signal encoding standard, such as the well-known MPEG-1 and MPEG-2 standards. In addition to raw video data, the encoder typically encodes instructions to the decoder as to how the raw video data should be interpreted and represented (i.e., displayed). For example, an encoded video stream may include instructions that a subsequent video picture is identical to a previous picture in a video stream. In this case, the encoded video stream can be further compressed, because the encoder need not send any raw video data for the subsequent video picture. When the decoder receives the instruction, the decoder will simply represent the subsequent picture using the same raw video data provided for the previous picture. Those skilled in the art will appreciate that such instructions can be provided in a variety of ways, including setting a flag or bit within a data stream.

Figure 5 is a block diagram depicting an exemplary selection of picture encoding modes in a GOP. As described above in connection with Figure 1, the video stream can be described in terms of I-pictures **503**, B-pictures **504**, and P-pictures **506**. A video stream can be represented by a series of groups of pictures (GOPs).

14

5   Each GOP begins with an I-Picture and includes one or more P-pictures and/or B-pictures. As described above, the I-Picture requires the most video data and is represented without reference to any other picture in the video stream. The P-Picture **506** can be represented in terms of differences with the I-Picture **502**. Likewise, the B-Picture **504** can be represented in terms of differences with the I-Picture **502** and/or

10   the P-Picture **506**. In conventional encoding methods, the size of the GOP **508** is arbitrarily set to a specific number of pictures. Consequently, during the encoding process, the first picture is classified as the I-Picture and is followed by a collection of P-pictures and B-pictures. When the predetermined number of pictures have been collected into a GOP, a new GOP can be started. The new GOP is started by

15   identifying a next picture as an I-Picture.

       In an exemplary embodiment of the present invention, the size of each GOP may be variable. In one embodiment, I-Frames coincide with scene changes in the input video stream. As is well known in the art, a scene change can be detected by significant changes and/or structural breakdown of motion vectors from one picture to

20   the next. Once a scene change has been detected, the picture following the scene change (i.e., first picture of the new scene) may be classified as an I-Picture.

       Figure 6 is a block diagram depicting an exemplary timeline comparing the occurrence of scene changes in a video stream with alternative GOP size formats. The video stream **600** is represented as a series of four scenes. Scene

25   changes occur at times **608, 610**, and **612**. In a conventional encoding system, the GOP is set at a constant number of frames, as depicted by GOP series **604**. Notably, the I-Frames in GOP format **604** occur at times **616, 618, 620**, and **622**. None of these times correspond with the times of the scene changes in the video stream **600**.

       The variable GOP format **602** is an exemplary embodiment of the

30   present invention. Typically, the I-Frames of the variable GOP format coincide with the scene changes in the video stream **600**. However, where a scene is sufficiently long, the variable GOP format **602** will default to a constant GOP size and insert an I-Picture as needed, as shown at time **606**. Consequently, some GOPs of the variable GOP format **602** will be longer than the typical size of constant GOP format **604**.

15

5   Other GOPs of the variable GOP format **602** (e.g., GOP **614**) will be significantly longer than the typical size of the constant GOP format **604**.

A major objective of the variable GOP format **602** of an exemplary embodiment of the present invention is to coincide I-pictures and scene changes. Because both I-pictures and scene changes require the most amount of video data

10   storage, the coincidence of these frames reduces the amount of data required to represent and encoded video stream. Another major objective of the variable GOP format **602** of an exemplary embodiment of the present invention is to maximize the benefit of novel adaptive bit allocation and conditional replenishment methods that are described in more detail in connection with Figures 8-12.

15

An Exemplary Method for Generating Variably-Sized Groups of Pictures

Figure 7 is a flowchart depicting an exemplary method for creating GOPs of varying sizes. The method begins at start block **700** and proceeds to step **702**. At step **702**, the first GOP is created and a first picture from an input video

20   stream is retrieved. The method proceeds to step **704**, wherein the first picture is classified as the I-Picture and is added to the first GOP.

The method proceeds from step **704** to decision block **706**. At decision block **706**, a determination is made as to whether more pictures exist in the input video stream. If a determination is made that more pictures exist in the video stream,

25   the method branches to step **710**. If, on the other hand, a determination is made that no more pictures exist in the video stream, the method branches to end block **708** and terminates.

At step **710**, the next picture from the video stream is retrieved. The method then proceeds to decision block **712**. At decision block **712**, a determination

30   is made as to whether the predefined GOP picture limit has been reached. As described above in connection with Figure 6, in the case where a scene is longer than the predefined GOP size, the method will created a new GOP rather than allow the variable GOP to reach an indefinite size. If the predefined GOP picture limit has been reached, the method branches to step **716** and a new GOP is started. If, on the other

16

5   hand, the standard GOP picture limit has not been reached, the method branches to decision block **714**.

At decision block **714**, a determination is made as to whether a scene change has been reached in the video stream. As described above, a scene change can be detected by various well-known means. If a scene change has been detected, the

10   method branches to step **716** and new GOP is started. If, on the other hand, a scene change has not been reached, the method branches to step **718** and the retrieved picture is added to the current GOP. The method proceeds from step **718** to decision block **706** and proceeds as described above.

Accordingly, pictures from an input video stream are added to a GOP

15   until either a scene change occurs or the predefined GOP size is reached. Exemplary GOP sizes range from a minimum of 15 frames to a maximum 60 frames. Those skilled in the art will appreciate that GOPs of widely varying sizes could be used within the scope of the present invention. As described above, the objective of the exemplary method is to coincide scene changes and I-Frames so as to minimize the

20   number I-Frames and scene change frames stored in an encoded video stream.

Figure 8 is a graph depicting a typical relationship between the bits generated by a conventional compression method and a conventional group of pictures. The graph **800** is divided into three groups of pictures (GOPs) **802, 804, 806**. Each GOP **802, 804, 806** begins with an I-picture **808, 810, 812**. As described above,

25   most conventional compression methods remove irrelevant, redundant, and/or expendable bits from a video stream. This is done by removing as much video data as possible from each picture in an input video stream. In addition, conventional compression methods encode pictures such that the content of the encoded pictures can be predicted from previous and/or subsequent pictures and the encoded video

30   stream. Accordingly, much of the video data for such predictable pictures can be eliminated from the encoded video stream, thereby further reducing the size of (i.e., further compressing) the encoded video stream. I-pictures **808, 810, 812**, however, are used to predict the video data content of other pictures (e.g., B-pictures, P-pictures) and typically contain more video data than other pictures in an encoded

35   video stream.

5   Referring again to Figure 8, it is apparent that for the I-pictures **808**, **810**, **812** more bits are generated during the compression process than for non-I-pictures **814**, **816**, **818**. As described above, conventional compression methods select pictures in an input video stream as I-pictures in an arbitrary fashion, based primarily on the number of pictures in a particular GOP. In an exemplary

10   embodiment of the present invention, I-pictures **808**, **810**, **812** can be selected to coincide with scene changes. Typically, scene-change pictures and I-pictures require the compression process to generate more bits than for non-scene change pictures or for non-I-pictures. By classifying scene-change pictures as I-pictures, an exemplary embodiment of the present invention reduces the overall number of bits generated by

15   the compression process. Because a large number of bits must be stored with an I-picture, regardless of the picture content, classifying scene-change pictures as I-pictures simply capitalizes on this feature to reduce the overall number of bits generated by the compression process.

           Figure 9 is a series of block diagrams and graphs comparing the

20   generated bit graph of a conventional compression method with a generated bit graph of an exemplary embodiment of the present invention. An input video stream is represented as a block diagram **900** divided into scenes. As described above, a conventional compression method divides groups of pictures on a fixed bases (i.e., the same number of pictures per group). A fixed-sized GOP structure is depicted as a

25   block diagram **904**. As described in connection with Figure 8, each GOP begins with an I-picture **910-916**. The fixed GOP Graph **908** has generated bit peaks that coincide with the I-frames **910-916** of each of the fixed-sized GOPs in the block diagram **904**. In addition, the fixed-sized GOP graph **908** also includes peaks coinciding with the scene changes between Scene 1 and Scene 2, between Scene 2 and Scene 3, and

30   between Scene 3 and Scene 4. Accordingly, the conventional, fixed-size GOP compression method generates output bit peaks for both I-pictures and scene-change pictures. Therefore, the bit budget for the remaining P-pictures and B-pictures is decreased. The encoding quality of the remaining P-pictures and B-pictures is, therefore, compromised or degraded.

18

The variable size GOP graph **906**, on the other hand, depicts output bit peaks coinciding primarily with scene changes in the input video stream **900**. Accordingly, the variable-sized GOP compression method of an exemplary embodiment of the present invention reduces the number of output bit peaks in the encoded video stream. More specifically, the variable-sized GOP compression method minimizes the number of double output bit peaks. These double peaks are present in the fixed-sized GOP graph **908** and are created when scene changes occur within a GOP, instead of coinciding with an I-picture of the GOP. As a result, the overall number of output bits generated by the fixed-sized GOP compression method is greater than the overall number of bits generated by the variable-sized GOP compression method of an exemplary embodiment of the present invention.

Accordingly, the exemplary compression method results in a smaller number of generated compression bits. This advantage provides various benefits to an encoding/decoding process. First, the resultant, smaller encoded video stream can be stored and/or transmitted in its smaller state, thereby conserving system resources. Alternatively, the encoding quality can be improved by re-allocating bits from smaller GOPs to larger GOPs. This is referred to as adaptive bit allocation, because the bit allocated to a given GOP can be adapted to the GOP size, which varies depending on the scene changes in the input video stream. This benefit is described in more detail in connection with Figure 10.

Exemplary Methods for Adaptive Bit Allocation

Figure 10a is a flow chart depicting an exemplary method for adaptively allocating bits among variable-sized groups of pictures (GOPs). In an exemplary embodiment of the present invention, bits can be allocated among the variable-sized GOPs. In addition, bits may be allocated among the pictures within a single GOP. These methods may be utilized individually or in concert to maximize the image quality of a compressed video stream and of the pictures within a GOP, while benefiting from the enhanced compression processes of exemplary embodiments of the present invention.

5   The method of Figure 10a begins at start block **1000** and proceeds to step **1002**. At step **1002**, the target bit number of a first GOP is determined. This step may be performed prior to encoding a GOP. For example, after an input stream has been segregated into GOPs, the GOPs may be stored in a buffer. Because the GOPs in the buffer may have different sizes (i.e., contain variable numbers of pictures), they

10   also may have different numbers of bits allocated thereto. The method of Figure 10a provides a means for adaptively allocating bits among GOPs, depending on the relative sizes of the GOPs.

          The method proceeds from step **1002** to step **1004**. At step **1004**, the number of bits actually generated for the pictures in the GOP is determined. The

15   method proceeds from step **1004** to decision block **1006**. At decision block **1006**, a determination is made as whether the bit size of the first GOP is less than the target bit number. If the GOP bit size is less than the target bit number, the method branches to step **1010**. If, on the other hand, the GOP size is not less than the target bit number, the method branches to end block **1016** and terminates.

20   At step **1010** the size and target bit number of a second GOP is determined. The method proceeds from step **1010** to step **1014**. At step **1014**, bits from the first GOP are allocated to the second GOP. That is, bits that would otherwise be assigned to the first GOP are reassigned to the second GOP, so that the quality of the second GOP is enhanced. As described above, the picture quality of the

25   encoded video stream is directly related to the bit rate of the encoded video stream. Accordingly, by reallocating bits between GOPs in a video stream, an exemplary embodiment of the present invention can maximize the quality of the GOPs having bit sizes larger than the target size, while retaining the picture quality of GOPs having bit sizes less than the target bit size. Conventional encoding methods cap the bit size of

30   any given GOP at the target bit size. Thus, for GOPs having a larger bit size, the picture quality is reduced as compared to those GOPs having smaller bit sizes.

          Figure 10b is a flow chart depicting an exemplary method for adaptively allocating bits among pictures within a GOP. In this embodiment of the present invention, bits can be adaptively allocated between pictures within a GOP.

35   For a GOP containing N-frames, N-1 bit values can be allocated to the non-I-picture

20

5   frames. The bit allocation can be based on a per-picture target bit size. The bits may

be allocated using the Root Mean Square (RMS) of the difference between the

successive frames. Preferably, the amount of bit allocation for the ith picture in a GOP

can be calculated as follows:

$$T_P(i) = \frac{R \times RMS(i)}{\sum_{l=1}^{N-1} RMS(l)}$$

10   where $T_P(i)$ represents the target bit rate for a current picture, R represents the target

bit rate for the remaining pictures in the GOP and RMS(i) represents the RMS value

of the difference between i^{th} picture and $i\text{-}1^{th}$ picture in the GOP. After encoding

each picture in the GOP, the target bit rate for the remaining pictures in the GOP (R)

can be updated by subtracting the number of actually generated bits for each picture.

15   When the number of bits that have actually been generated for all of the pictures in

the GOP is less than the target bit rate, then the bits may be made available for

allocation to pictures in other GOPs. In this embodiment of the present invention, bits

can be allocated on a picture-by-picture basis within a GOP, so as to maximize the

picture quality on a picture-by-picture basis.

20   Turning now to Figure 10b, an exemplary method is depicted, wherein

bits are adaptively allocated among the pictures in a GOP. The method of Figure 10b

may be implemented at the time that the picture size (i.e., number of pictures) for a

subject (current) GOP has been defined, for example, by the Picture Grouping Module

316 described in connection with Figure 3. The method begins at start block **1050**

25   and proceeds to step **1052**. At step **1052**, the size of the GOP is determined. This

step may be performed by the Picture Grouping Module **316** or the pictures in the

GOP may simply be re-counted. The method then proceeds to step **1054**, wherein the

target bit number for the current GOP is determined. Typically, a compression

process is implemented for a particular application wherein an overall bit rate is

30   predetermined. Those skilled in the art will appreciate that this overall bit rate may be

used to determine a bit rate on a per-picture basis.

The method proceeds from step **1054** to step **1056**. At step **1056**, the

Root Mean Square (RMS) of the difference between a current picture and a previous

21

5   picture is determined. Initially, the current picture will be the first picture in the GOP. This step can be performed using the formula described above. The method then proceeds to step **1058**, wherein the appropriate number of bits is actually allocated to the current picture. The method then proceeds to decision block **1060**, wherein a determination is made as to whether all of the pictures in the GOP have been encoded.

10   If a determination is made that all of the pictures in the GOP have been encoded, the method branches to decision block **1062**. If, on the other hand, a determination is made that all of the pictures in the GOP have not been encoded, the method branches to step **1068**.

            At step **1068**, the current picture is incremented. That is, the next

15   picture in the GOP is identified for bit allocation consideration. The method then proceeds to step **1056** and proceeds as described above. Returning now to decision block **1062**, a determination is made as to whether the number of bits actually generated by encoding all of the pictures in the GOP is less than the target bit total for all of the pictures in the GOP. If the number of bits actually generated by encoding

20   the pictures in the GOP is not less than the target bit total for all of the pictures in the GOP, then the method branches to end block **1066** and terminates. If, on the other hand, the number of bits actually allocated to the pictures in the GOP is less than the target bit total for all of the pictures in the GOP, then the method branches to step **1064**. At step **1064**, the remaining bits (not allocated) are made available to the next

25   GOP (or some other subsequently processed GOP) to be considered for bit allocation. The method proceeds from step **1064** to end block **1066** and terminates.

            Accordingly, the method efficiently allocates bits among pictures within a GOP. Where a surplus of bits exists, the method can make those bits available for subsequent GOPs, for which such a surplus does not exist. Because the

30   GOP size is variable in accordance with exemplary embodiments of the present invention, this bit allocation method capitalizes on bit surpluses that are created by using variable GOP sizes. The described bit allocation methods can be used to significantly improve the output quality of an encoding system by efficiently using bits that might otherwise be imprudently allocated.

35

22

5 An Exemplary Method of Conditional Replenishment

Conditional replenishment is a well-known aspect of conventional compression methods. Generally conditional replenishment refers to the elimination of redundant video data in a condition wherein video data remains unchanged between successive pictures in a GOP. More specifically, conditional replenishment

10 is a method of "re-using" (i.e., replenishing) previously encoded video data to populate an area of a video image that is unchanged from a previous video image. When possible, such replenishment reduces the amount of new video data that must be encoded, therefore reducing the output bit rate and increasing output bit quality.

Because successive pictures within an exemplary variable-sized GOP

15 are typically members of the same scene in an input video stream, the opportunity for conditional replenishment is increased with a given GOP. Accordingly, the scene-oriented GOP sizing of exemplary embodiments of the present invention enhance the performance of conventional replenishment methods. In addition, because of the similarity between successive pictures in a given GOP, a novel variation of

20 conditional replenishment is applied in an exemplary embodiment of the present invention to further enhance video stream compression.

Figure 11 is a simplified illustration depicting successive pictures in an exemplary GOP divided into macroblocks. Picture **1100** is divided into macroblocks **1102-1114**. Likewise, picture **1150** is divided into macroblocks **1152-1164**.

25 Although the image in picture **1100** is different than the image in picture **1150**, only certain macroblocks are different. Specifically, macroblocks **1102-1110** of picture **1100** are different than macroblocks **1152-1160** of picture **1150**. On the other hand macroblocks **1112-1114** of picture **1100** are identical to macroblocks **1162-1164** of picture **1150**. Accordingly, picture **1150** may be represented (i.e., encoded) as being

30 identical to picture **1100**, except for changes to macroblocks **1152-1160**.

When it is determined that a difference exists between corresponding coded pixels in the macroblock, the differences can be stored or transmitted in connection with the corresponding picture. If, on the other hand, it is determined that no difference exists between corresponding coded pixels, then a flag can be set to

35 indicate (or other instruction provided) that the pixel from the previous picture can be

5   used, thereby eliminating a need to store additional information for the successive picture graph.

In conventional conditional replenishment, the replenishment condition is determined by examining the results of the encoding process. If the encoding results (quantized DCT coefficients) are exactly same between the macroblocks of

10   current frame and previous frame, replenishment is used. In an exemplary embodiment of the present invention, on the other hand, conditional replenishment is performed intelligently by the encoder, based on a calculation of relevant criteria. Accordingly, if the encoder does not detect a replenishment condition, any change detected between corresponding macroblocks in successive pictures may be stored or

15   transmitted. On the other hand, when the encoder detects a replenishment condition, then an instruction and/or flag can be used to indicate that the macroblock should be replenished using the video data from the previous picture.

Advantageously, conditional replenishment on a macroblock basis enables noise reduction in an encoded video stream. When an encoded video stream

20   is decoded, noise is commonly detectable in a displayed video stream as a flickering or otherwise perceivable image. Often, such noise is more perceivable when it occurs in a background region (i.e., a region of substantially constant image intensity). In an exemplary embodiment of the present invention, conditional replenishment is processed on a macroblock basis, utilizing 2-part criteria and selectable thresholds for

25   modifying the criterion . As a result, slight differences resulting from noise in a particular macroblock can be muted (i.e., filtered). The first criterion can be used to determine the differences between an original macroblock and a previous macroblock. This criterion, C1, is given by the expression:

30

$$C1 = \sqrt{\frac{1}{256} \sum_{i=1}^{16} \sum_{j=1}^{16} (org(i,j) - prev(i,j))^2}$$

where $org(i,j)$ represents the ith and j^{th} pixel of the original (subject) macroblock and $prev(i,j)$ represents the ith and j^{th} pixel of original macroblock of the previous frame.

24

5   The second criterion, may be used to evaluate the effect of the
decoder, by reference to the original macroblock. The second criterion, C2, is given
by the expression:

$$C2 = \sqrt{\frac{1}{256} \sum_{i=1}^{16} \sum_{j=1}^{16} (org(i,j) - coded(i,j))^2}$$

10

where $org(i,j)$ represents the ith and j^{th} pixel of the original (subject) macroblock and
$coded(i,j)$ represents the ith and j^{th} pixel of the decoded macroblock of the previous
frame. Criterion 1 is the measurement of similarity of the corresponding macroblocks
of the current frame and the previous frame. Criterion 2 is for double check of the
15   similarity with the decoded macroblock.

In addition, threshold values may be selected for the two criteria, to set
the sensitivity of the conditional replenishment process. Alternatively, the threshold
may be automatically set such that it is adaptive to a particular bit rate. The following
table provides an exemplary relationship between bit rate and Criterion 1 (C1)
20   threshold values.

BIT RATE	THRESHOLD 1
greater than 400 k	8
300 k - 400 k	11
200 k - 300 k	13
110 k - 200 k	14
less than 110 k	15

Similarly, the threshold value for Criterion 2 may be set manually or automatically
(an exemplary value for Threshold 2 is 8). By applying the 2-part criteria in
25   conjunction with the threshold values, the macroblock-based conditional
replenishment method of an exemplary embodiment of present invention can be used
and fine-tuned to reduce noise in a displayed video stream.

5   Figure 12 is a flowchart depicting an exemplary method for performing conditional replenishment on a macroblock-basis. The method of Figure 12 begins at start block **1200** and proceeds to step **1202**, wherein a first macroblock is compared to a second macroblock. The method then proceeds to decision block **1204**, wherein a determination is made as to whether Criterion 1 (C1) is less than Threshold 1. If at

10   decision block **1204**, a determination is made that Criterion 1 is not less than Threshold 1, the method branches to step **1210**. At step **1210**, a flag can be set for an instruction providing that the second macroblock should be encoded using the data from the first macroblock, rather than simply replenished. The method proceeds from **1210** to end block **1212** and terminates.

15   Returning now to decision block **1204**, if a determination is made that the Criterion 1 is less than Threshold 1, the method branches to decision block **1206**. At decision block **1206** a determination is made as to whether Criterion 2 is less than Threshold 2. If a determination is made at decision block **1206** that Criterion 2 is not less than the Threshold 2, the method branches to step **1210** and proceeds as described

20   above. If on the other hand, a determination is made at decision block **1206** that Criterion 2 is less than Threshold 2, the method branches to step **1208**. At step **1208** the replenishment flag is set for the second macroblock. The method proceeds from step **1208** to step **1212** and ends.

Accordingly, the method of Figure 12 can be used to utilize selectable

25   criteria to reduce the encoding, decoding and display of noise. The replenishment of an exemplary embodiment of the present invention, thus, can be used to filter noise from a displayed video stream. Those skilled in the art will appreciate that various criteria and/threshold values may be used within the scope of the described embodiments of the present invention.

30

An Exemplary Method for Selecting an Asynchronous Sampling Technique

To maximize the quality of compressed video at a low bit rate (e.g., less than 128 kbps), it may be useful to sample the video at optimum points in time and space. Sampling is roughly defined as the determination of which pictures in a

35   video stream will be encoded as I-pictures, B-pictures, and P-pictures. Generally,

26

optimum sampling can be non-uniform (asynchronous) in one or both of the space and time domains. Various asynchronous techniques are well known to those skilled in the art and can be used to implement various embodiments of the present invention. In an exemplary embodiment of the present invention, an analysis-by-synthesis method of selecting an asynchronous sampling technique is provided. In the exemplary analysis-by-synthesis method, separately encoded candidate streams are generated using various sampling methods. Once generated, the separate candidate streams can be compared on virtually any basis to determine, for example, which has the best bit rate and signal quality characteristics. The best candidate stream can be selected and designated as the output video stream. The selected sampling method can be identified to the receiver (decoder) with a small overhead. For example, by using a codebook or dictionary of 16 possible sampling techniques, only 4 bits of overhead are needed to signify the selection. The codebook could be either predetermined or generated adaptively (and automatically) over time, based on criteria including extrapolation from a recent history of optimum sampling.

Figure 13 is a flowchart depicting an exemplary method for generating and selecting between two sampling methods. Those skilled in the art will appreciate that any number of sampling methods could be used and evaluated within the scope of the present invention. It also will be appreciated that the generation of multiple candidate streams creates overhead as described above, and that the exemplary sampling selection method may be more easily applied to one-way communications (e.g., video streaming), than to two-way communications (video teleconferencing).

The method of Figure 13 begins at start block **1300** and proceeds to step **1302**. At step **1302**, a first input video stream is encoded using a first sampling technique. The method then proceeds to step **1304**. At step **1304**, a second input stream is encoded using a second sampling technique. The method then proceeds to step **1306**, wherein the encoded candidate video streams are compared. This comparison could be based on various characteristics of the candidate video streams. However, it is preferable that the characteristics are perceptually meaningful characteristics. An exemplary characteristic is the signal-to-noise-ratio of each encoded candidate video stream, as compared to the original uncompressed signal.

27

5   The method proceeds from step **1306** to decision block **1308**. At decision block **1308**, a determination is made as to whether the signal-to-noise-ratio (SNR) for the first stream is higher than the SNR for the second stream. If the SNR for the first stream is better than the SNR for the second stream, then the method branches to step **1310**. At step **1310**, the first stream is output. Returning to decision

10   block **1308**, if the SNR for the second stream is better than the SNR for the first stream, then the method branches to step **1312**. At step **1312**, the second stream is output. Accordingly, the encoded candidate streams having been encoded using different sampling techniques are compared and the best stream is output, for example, from an encoding system, together with the overhead information that

15   signifies the corresponding sampling method..

          Although the present invention has been described in connection with various exemplary embodiments, those of ordinary skill in the art will understand that many modifications can be made thereto within the scope of the claims that follow. Accordingly, it is not intended that the scope of the invention in any way be limited

20   by the above description, but instead be determined entirely by reference to the claims that follow.